**基于改进粒子群优化算法的车辆路径问题研究**

**Lin Wang**

Chang'an University, School of Automobile

Middle-section of Nan'er Huan Road, Xi'an, China

[wangling\_fighting@163.com](mailto:wangling_fighting@163.com)

**Guoliang Shen**

Chang'an University, School of Electronics and Control Engineering

Middle-section of Nan'er Huan Road, Xi'an, China

[15771772135@163.com](mailto:15771772135@163.com)

**Shifeng Niu**

Chang'an University, School of Automobile

Middle-section of Nan'er Huan Road, Xi'an, China

nsf530@163.com

**摘要**

为了提高粒子群优化的优化能力，我们提出了一种改进的粒子群优化算法，并将其应用于解决车辆路径问题。基于线性和非线性惯性权重组合的方法，自适应粒子群优化算法有助于在前期进行局部搜索，在后期进行全局搜索。结合模拟退火算法，改进的粒子群优化算法可以避免陷入局部最优解和与最优解的差距过大。本文构建了车辆路径模型和改进的粒子群优化算法对车辆路径问题进行优化。结果表明，相较于遗传算法，改进的粒子群优化算法可以快速有效地获得车辆路径问题的最优解，表明改进的粒子群优化算法是更有效的解决车辆路径问题的方法。

**关键词**：粒子群优化 车辆路径问题 惯性权重 模拟退火算法

**1 引言**

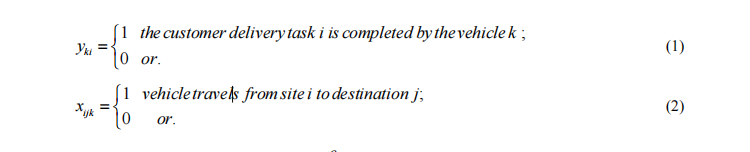
车辆路径问题（VRP）最初由Dantzig和Ramser于1959年提出[1]，是一种组合优化和整数规划问题。它指的是在特定约束条件下，一系列配送点（或接收点）形成合适的行驶路径，让车辆按顺序通过完成确定的目标（如最短距离、最少成本等）。VRP是NPC问题，存在于生活的许多方面，是运输、配送和物流领域的重要问题。解决VRP的主要方法是启发式算法和精确算法。在启发式算法中，遗传算法、禁忌搜索算法、模拟退火算法等经常用于解决VRP，并取得了良好的结果。

粒子群优化（PSO）[2]是一种仿生算法，模拟了鸟类飞行的过程，具有个体数量少、计算简单和健壮性好等优点。PSO在各种多维连续空间优化问题中取得了优秀的表现[3]。本文提出了一种改进的粒子群优化算法（IPSO）。首先，我们使用基于线性和非线性惯性权重组合的自适应粒子群优化算法，有利于在前期进行局部搜索和后期进行全局搜索。然后，为了避免粒子群优化陷入局部最优解，引入了遗传算法中的交叉思想。IPSO具有更快的收敛速度，能够快速搜索全局最优解，并提高了算法的效率。

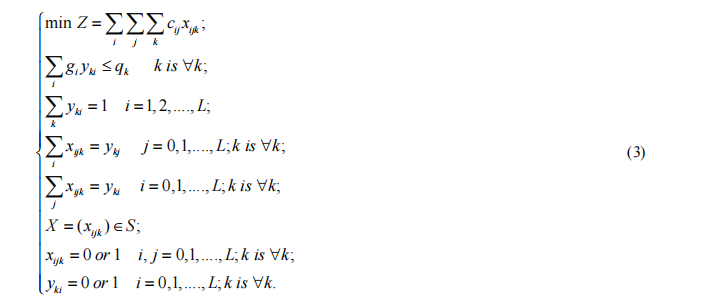
**2 VRP问题的数学描述**

VRP的一般描述如下：一个物流中心有总共K辆车，每辆车的容量为qk(k=1,2,...,K)。现在有L个需要完成的客户送货任务，用1,2,...,L表示这些任务。第i个客户送货车辆的容量为gi(i=1,2,...,L)，且max gi ≤ max qk，最终得到完成任务的最短路径。

本文采用的数学模型是文献[4]中提出的模型。物流中心编号表示为0，客户送货任务编号表示为1,2,...,L。送货任务和物流中心分别表示为i(i=0,1,2,...,L)。定义如下：



运输货物的成本在这里表示为cij，表示车辆从起点i到目的地j的成本。这里的cij也可以代表其他意义，比如距离、时间等。车辆优化所需的条件由以下表达式表示：



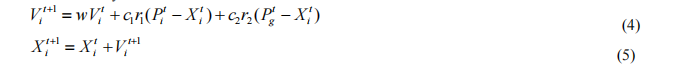
如公式所示，运输所需的条件包括：（1）每个客户送货任务都有一个运输车辆；（2）每个客户送货任务都有一个车辆来完成；（3）每个客户送货任务的总需求量不会超过每辆车的总容量；（4）车辆行驶距离达到最小值。

**3 改进粒子群优化算法**

IPSO是一种群体智能方法，由Kennedy和Eberhart于1995年提出[5]，该算法的思想源于对鸟类捕食行为的模仿[2]。每个粒子的状态对应于每个优化问题的解决方案，并且每个粒子都有自己的，由优化函数确定的自适应值。具有最高自适应值的粒子是最优粒子。每个粒子在每一代都在寻找其最佳自适应值，最终找到了自己和整个群体的最佳自适应值。

**3.1 简单粒子群优化算法**

假设搜索空间的维度为n，总粒子数量为m，迭代次数为t。第i个粒子的位置由向量表示，表示为 = (xi1, xi2,..., xin) 。速度由向量表示，表示为 = (vi1, vi2,..., vin) 。到目前为止第i个粒子搜索的最佳位置表示为 (pi1, pi2,...,pin) ，对应的解是最优解。第g个粒子搜索的位置是到目前为止整个种群历史中的最佳位置，并由向量表示为 = (pg1, pg2,..., pgn) 。然后，对其位置和速度进行更新，更新表达式如下：



其中，1 ≤ i ≤ m 且搜索在n维空间中进行； c1 和 c2 是非负常数，通常称为加速因子或学习因子； w 是惯性权重； r1 和 r2 是在[0,1]之间的随机数； t 表示当前代的迭代次数； t+1 表示下一代的迭代次数。

由公式（4）和公式（5）可见，惯性权重 w 反映了上一代速度对当前一代速度的影响。学习因子 c1 反映了自身搜索的最佳位置对粒子在空间中搜索的影响；学习因子 c2 反映了粒子群搜索的最佳位置对粒子在空间中搜索的影响。学习因子 c1 和 c2 的值应该保持在一个相对平衡的范围内。一般将 c1 和 c2 的值设置为 c1 = c2 = 2。在搜索过程中，粒子速度和位置的每个维度应该有一个限制，这里定义的速度限制为 [-V maxd, V maxd]，位置限制为 [-X maxd, X maxd]。当粒子的速度和位置超出搜索过程中的限制时，粒子将被视为达到了边界值。在分析了PSO的参数之后，PSO的收敛参数条件由Maurice[6]给出。

**3.2 改进的粒子群优化算法**

PSO的主要改进如下：

（1）通过结合线性和非线性方法改进惯性权重。惯性权重可以平衡上一代的速度与当前代的速度，在迭代过程中展示PSO算法的搜索能力。随着迭代次数的增加，在前期使用线性递减函数来控制惯性权重，有利于全局搜索；在后期采用非线性递减函数来控制惯性权重，有利于局部搜索。

（2）结合模拟退火算法。PSO简单、快速且易于实现，但容易出现早熟收敛并且容易陷入局部最优解。模拟退火算法具有较好的全局搜索能力，为了进一步提高算法的有效性，PSO通过结合模拟退火算法进行改进，以避免局部最优和偏离最优解。

线性和非线性惯性权重的改进表达式如下：



其中，Wmax是最大的惯性权重，即惯性权重的初始值；Wmin是最小的惯性权重，即惯性权重的最终值；h是迭代次数，其范围是0 ≤ h ≤ T；T是最大迭代次数；W'是迭代过程中迭代次数为t = h时的惯性权重值；h1(t)是迭代过程中迭代次数为t ≤ h时的惯性权重值；h2(t)是迭代过程中迭代次数为t ≥ h时的惯性权重值。

在使用线性和非线性组合的惯性权重时，粒子在初始阶段t ≤ h根据线性递减的惯性权重进行搜索。当t = h时，记录此时的惯性权重值，并将线性递减的惯性权重转换为非线性递减的惯性权重。然后，根据非线性递减的惯性权重进行搜索。

**3.2.2 线性权重和非线性权重的结合**

模拟退火算法源于固体退火原理。它最早由N. Metropolis等人在1953年提出，并在1983年由S. Kirkpatrick引入到组合优化领域。

在组合优化领域，固体的退火过程可以形式化为：固体的内部能量等同于组合优化问题中的评估函数；固体的状态等同于问题的解；固体的最小内部能量等同于问题的最优解。

假设 C(st) 是评估函数，其中 st 是当前状态。当前温度是 T。当能量差异为 ΔE 时，温度降低的概率是 P(ΔE)。模拟退火算法的数学描述过程如下所示：

(1) 如果 C(st+1) ≥ C(st)，即下一个状态的评估函数大于或等于当前状态的评估函数值时，粒子总是接受移动；

(2) 如果 C(st+1) < (st)，即下一个状态的评估函数小于当前状态的评估函数值时，粒子将以一定的概率接受移动，而这个概率会随着迭代次数的增加而减小。

概率的数学表示形式为：



其中，K 是常数，而 ΔE< 0。

由公式（7）可见，随着温度 T 的增加，降低能量差异ΔE温度的概率增加。相反，温度越低，冷却的概率越小。

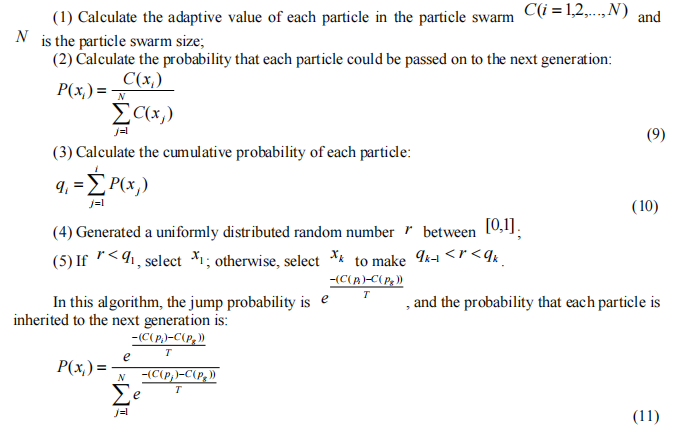
因此，在粒子群寻找最优解的过程中，如果下一个状态的解优于当前解（下一个状态的适应值大于当前适应值），则使用下一个状态的解替换当前解，即更新当前解。如果下一个状态的解劣于当前解，则根据模拟退火算法的原则，以一定概率接受下一个状态的解，使其有一定概率作为当前解。

在应用模拟退火算法后，粒子在搜索时会以较低的概率接受适应值较低的粒子。因此，初始温度应该足够大，以确保粒子在初始阶段有强大的初始跳跃。这里的初始温度定义为：



其中，C(pb1)是在定义初始随机粒子时最佳粒子的评估函数值；C(pw1)是在定义初始随机粒子时最差粒子的评估函数值；ps是在初始温度下最佳粒子被最差粒子替换的概率。

轮盘赌算法也称为比例选择方法。其基本思想是每个个体被选中的概率与其适应值成正比。具体的计算过程如下：



因此，根据概率 P（xi），使用轮盘策略来确定更多的当前粒子是否替换当前粒子。在添加模拟退火算法后，具体的搜索步骤设计如下：

(1) 初始化粒子群。定义粒子群数量为 m，搜索空间维度为 d。定义代数初始迭代为 t = 1，最大迭代次数为 T。定义随机参数 r1 和 r2；定义学习因子为 c1 和 c2。初始惯性权重定义为 Wmax，并终止惯性权重 Wmin。给出每个粒子的初始位置 Xit 和初始速度 Vit，定义初始温度值 T0。

(2) 根据给定粒子的初始位置和初始速度，计算粒子群中每个粒子的初始适应值。

(3) 根据公式（6）计算惯性权重。

(4) 根据公式（4）、（5）更新每个粒子的下一代位置和速度。

(5) 根据更新后粒子的速度和位置计算适应值，并确定下一代粒子的适应值和当前适应值。如果下一代粒子的适应值优于当前适应值，则将下一代粒子的适应值作为粒子的最佳位置pit。否则，使用轮盘策略选择部分粒子并接受更多当前的粒子进行更新，其余粒子保持不变。

(6) 降低温度。温度的逆转函数选择为 Tk = λTk−1，其中 λ 是常数。

(7) 转到步骤 (3) 直到达到最大迭代次数。如果满足条件，则继续下一步。

(8) 输出迭代后粒子的最优位置和速度。

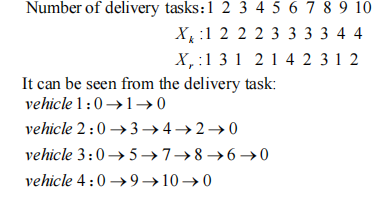
**4 应用**

**4.1 VRP模型**

参考文献[8]中的思想，建立了VRP模型如下：

建立了车辆路线维度，设置为2L维度。中央仓库对每个客户进行分配，因此交付过程涉及到分配目标任务的车辆选择以及每个任务车辆的交付顺序的选择。因此，对于车辆编号 k 和配送顺序 r，将 L 个交付任务分成两个 L 维的位置向量：Xk（交付目标，任务车辆选择）和 Xr（每个任务车辆的配送顺序）。相应的速度向量分别为 Vk 和 Vr。

例如，有10个需要完成的客户交付任务。仓库由中央仓库发货，中央仓库有4辆车，对应的位置向量可以表示为：



**4.2 算法实现**

(1) 初始化粒子群。将 Xk 的值定义为介于 1 到 k 之间的整数。 Xr 的值为介于 1 到 L 之间的实数。 Vk 的值为介于 -(k-1) 到 (k-1) 之间的整数。定义参数 r1、r2、c1 和 c2。定义初始迭代代数为 t = 1，最大迭代次数为 T。

(2) 计算粒子群中每个粒子的初始适应值。根据给定粒子的初始位置和初始速度，计算每个粒子的初始适应值。

(3) 根据公式（6）计算惯性权重。

(4) 根据公式（4）和（5）更新下一代每个粒子的速度和位置。

(5) 根据更新后粒子的速度和位置计算适应值，并确定下一代粒子的适应值大小和当前适应值。如果下一代粒子的适应值优于当前适应值，则将下一代粒子的适应值作为粒子的最佳位置 pit 。否则，使用轮盘策略选择部分粒子并接受更多当前的粒子进行更新，其余粒子保持不变。

(6) 降低温度。选择温度反转函数为 Tk = λ Tk-1，其中 λ 是常数。

(7) 转至步骤（3），直到达到最大迭代次数。如果满足条件，则继续下一步。

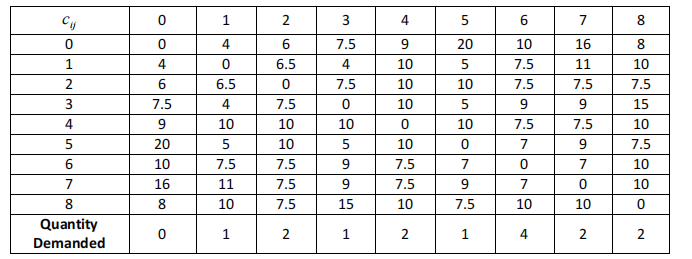
(8) 输出迭代后的粒子的最佳位置和速度。

**4.3 仿真和结果分析**

本文模拟并比较了文献[9][10]中描述的案例与文献[9][10]中提出的算法的结果。案例如下：

一个物流中心有2辆车，每辆车的载重量为8吨。有8个客户配送任务需要完成，每个客户之间的距离和需求量如表1所示。最终获得了解决问题的最短路径。在表1中，cij 表示物流中心与每个客户之间的距离，需求量的单位为吨。

表1 每个客户的需求量和物流中心与每个客户之间的距离



算法参数定义如下：

学习因子选为 c1 = c2 = 1。

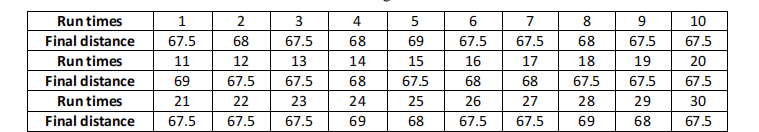
r1 和 r2 是在 [0, 1] 区间内的随机数。

预阶段的最大迭代次数为 T1 = 500，后续阶段的最大迭代次数为 T2 = 500。

在公式（6）中，预阶段的参数定义为 h = T1 / 2，h1(t) = t / T1 和 h2(t) = e^(-16(T1-t)^3) / T1^3，后续阶段的参数定义为 h = T2 / 2，h1(t) = t / T2 和 h2(t) = e^(-16(T1-t)^3) / T1^3。其中，λ 被定义为 λ = 0.95。

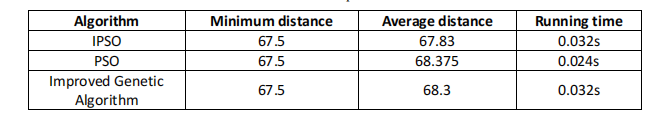
本文使用了IPSO，运行结果如表2所示。

表2 IPSO的运行结果



每辆车在交付任务中的路线顺序分别是0→4→7→6→0和0→1→3→5→8→2→0。将IPSO的结果与PSO和改进的遗传算法的结果进行比较。

表3 参数表



从表3可以看出，在IPSO运行30次的前提下，平均距离优于其他两种算法，并且运行时间更短。可以看出，IPSO在VRP中取得了良好的结果。与PSO和改进的遗传算法相比，IPSO具有高效率、良好的优化效果、高搜索结果和高质量的优势。因此，IPSO对于VRP是有帮助的。

**5 总结**

为了克服收敛速度慢和陷入局部最优解的缺点，本文提出了IPSO算法。将线性和非线性惯性权重的组合引入PSO中，改进了预阶段的局部搜索速度和后续阶段的全局搜索速度。通过有效地应用遗传算法进一步改进了PSO，提高了准确性和速度。示例显示，IPSO能够在相对短的时间内搜索到最佳路径并获得最短距离。与PSO和改进的遗传算法相比，结果表明IPSO能够提高搜索最佳路径的成功率，是更好、更有效的解决VRP的方法。

**参考文献**

[1] Dantzig, George Bernard; Ramser, John Hubert (1959). The Truck Dispatching Problem [C]. Management Science. 6 (1): 80–91.

[2] Kennedy J, Eberhart R C. Particle swarm optimization[A]．Proc．IEEE International Conference on Neural Networks, IV[C]. Piscaraway, NJ: IEEE Service Center, 1995. 1942-1948．

[3] Eberhart R C, Shi Y. Particle Swarm Optimization: Developments, Applications and Resources[C]. Proc. Congress on Evolutionary Computation 2001. Piscataway, NJ: IEEE Press，2001.81-86.

[4] Li Jun, Guo gang. Theory and method of vehicle optimal scheduling for logistics distribution [M]. Beijing: China Materials Press,.2001.76-77.

[5] Eberhart R C, Kennedy J. A new using particle swarm theory[C]. Proceedings of the 6th international Symposiumon Micro Machineand Human Science.Piscataway,N.J. ,USA:IEEE Service Center, 1995: 39-43.

[6] Steinbrunn M, Moerkotte G, Kemper A. Heuristic and Ran2 domized Optimization for the Join Ordering Problem[J ]. The VLDB Journal , 1997, 6 (3): 8-17.

[7] Shi Y, Eberhart R C. Empirical study of particle swarm optimization[A]. Proceedings of the 1999 Congress on Evolutionary Computation[C]. Piscataway, NJ: IEEE Service Center,1999. 1945- 1950

[8] Shi Y, Eberhart R C. Empirical study of particle swarm optimization[A]. Proceedings of the 1999 Congress on Evolutionary Computation[C]. Piscataway, NJ: IEEE Service Center,1999. 1945- 1950.

[9] Xiao Jianmei, Li Junjun, Wang Xihuai. Improved particle swarm optimization algorithm for vehicle routing problem. Computer integrated manufacturing system, 2005, 11(4):577-581.

[10] Zhang Yuxing, Fan Jianhua, Xu Jiangang, Chen Dongsheng. Improved genetic algorithm for vehicle routing problem. Journal of Tianjin University of Technology, 2006, 22 (5),:79 - 82.